LA-UR-13-22024
Approved for public release; distribution is unlimited.

Title: Small File Aggregation with PLFS

Author(s): Bonnie, David J.
Torres, Aaron G.

Intended for: Local Presentation

Small File Aggregation with PLFS

Aaron Torres

David Bonnie

Los Alamos National Laboratory

{agtorre,dbonnie}@lanl.gov

Abstract

- Today's computational science demands have resulted in ever larger parallel computers, and storage systems have grown to match these demands. Parallel file systems used in this environment are increasingly specialized to extract the highest possible performance for large I/O operations, at the expense of other potential workloads. While some applications have adapted to I/O best practices and can obtain good performance on these systems, the natural I/O patterns of many applications result in the generation of a huge number of small files, the creation of which is poorly served by current parallel file systems at very large scale. This paper describes a new technique for optimizing small file access in parallel file systems for these very large scale systems. The idea is to use a virtual parallel log-structure file system on the compute nodes in order to aggregate large numbers of small files in compute node memory and then stream their data sequentially to a much smaller number of physical files on an underlying parallel file system. The technique is implemented and evaluated using PLFS as the aggregating middleware. We evaluate our system with micro-benchmarks on a local OSX filesystem and with an MPI extension of the standard Postmark to provide results at scale on both Lustre and PanFS parallel filesystems. We observe as much as a 33x improvement in small file create rates on a single host, and 30x improvement in small file write rates, compared to a baseline Lustre configuration on a leadership computing platform using 16,384 cores and achieve an unprecedented create rate of 200 million files per second.

Figures

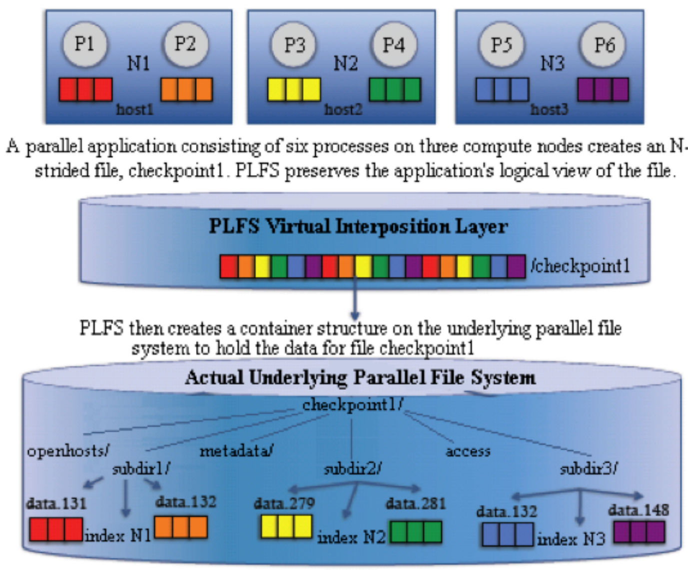


Figure 1: PLFS Data Reorganization

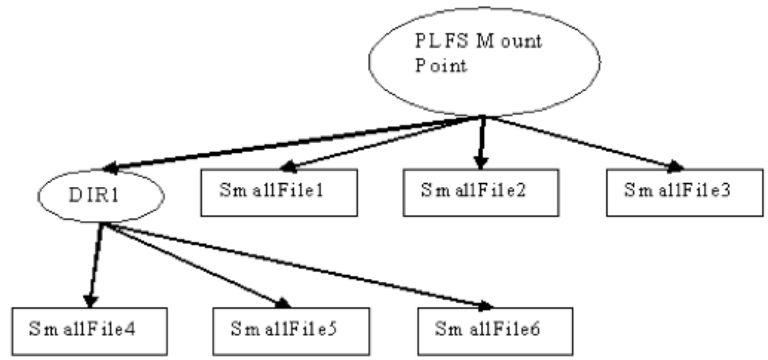


Figure 2: Logical View of the PLFS File System

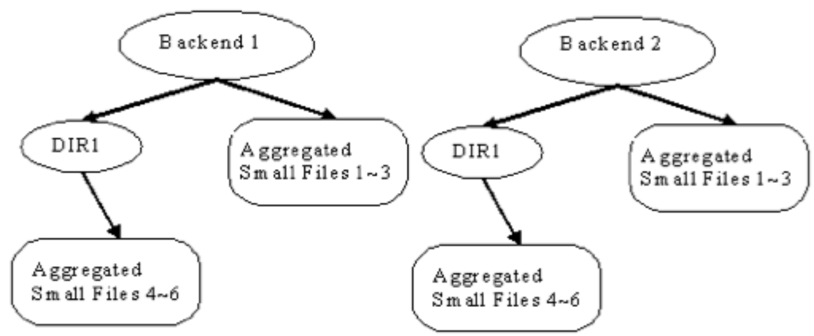


Figure 3: Physical View of the PLFS Backends

Figures

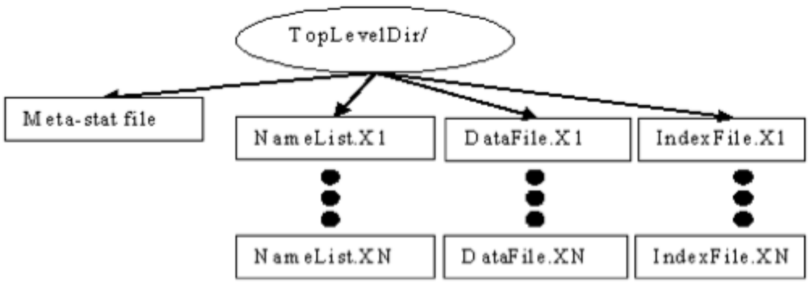


Figure 4: The Structure of Aggregated Small Files

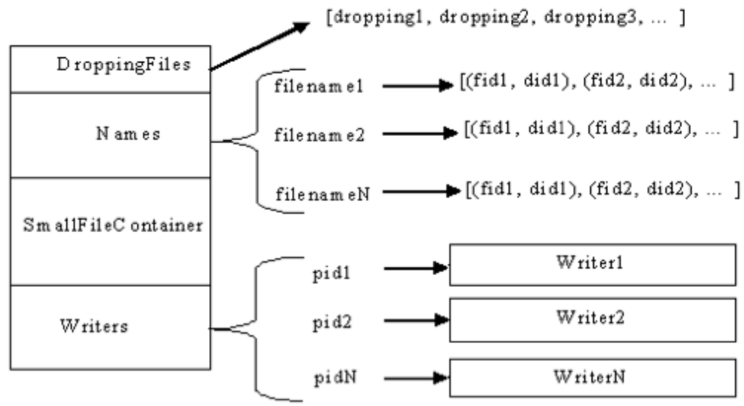


Figure 6: The SmallFileContainer

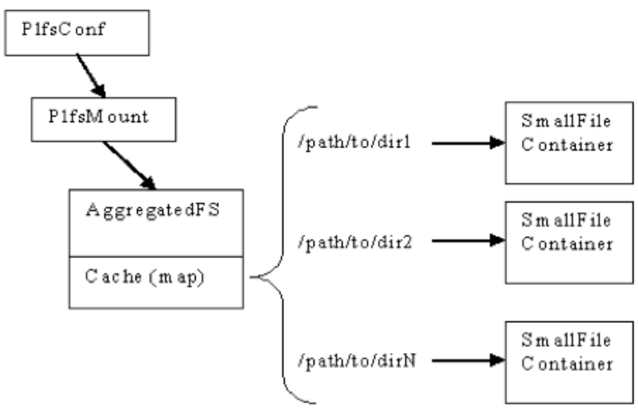


Figure 5: The Small File Cache in PLFS

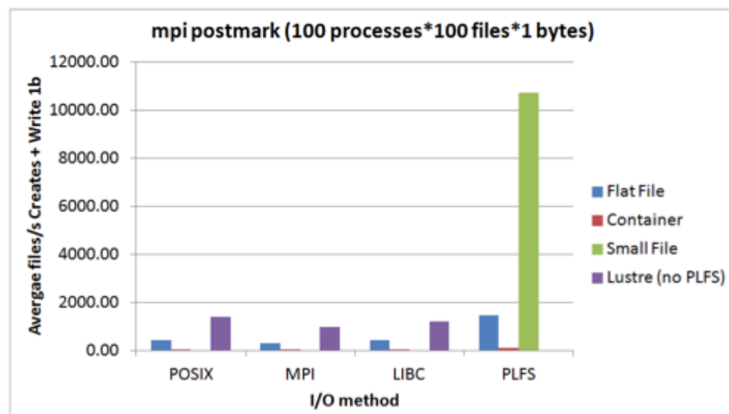


Figure 7: Postmark results on Lustre for different I/O methods

Figures

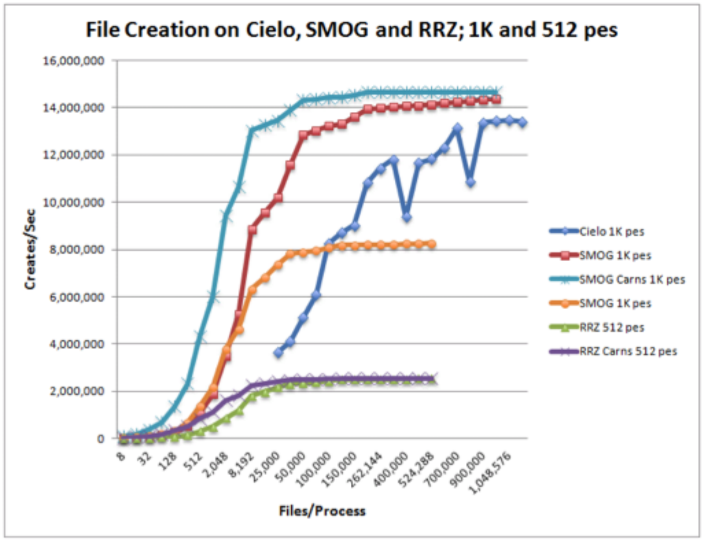
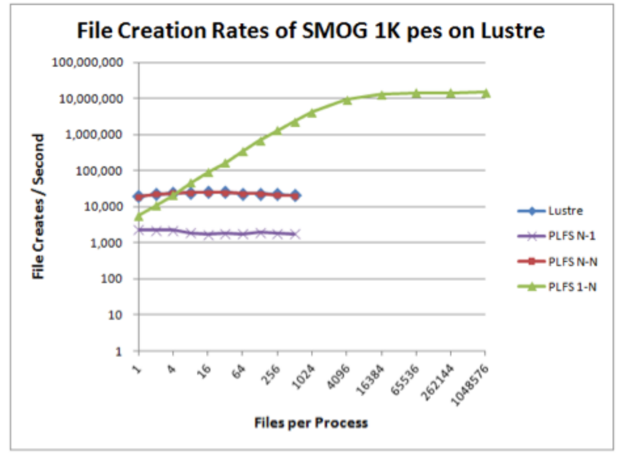
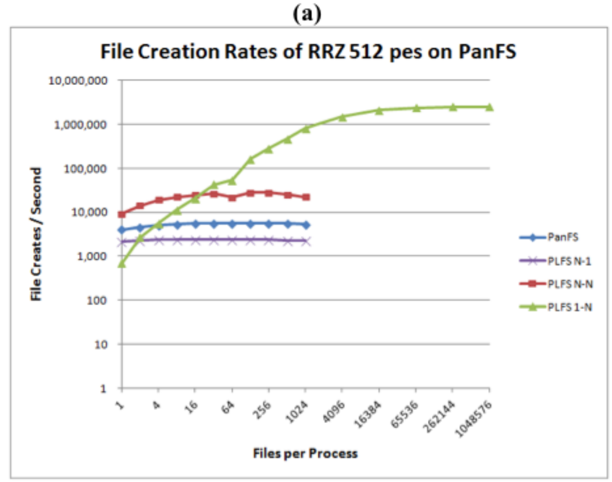


Figure 8: PLFS 1-N creates performance on 3 systems



(a)



(b)

Figure 9: PLFS Small Files and Container creates/s on (a) Lustre and (b) PanFS

Figures

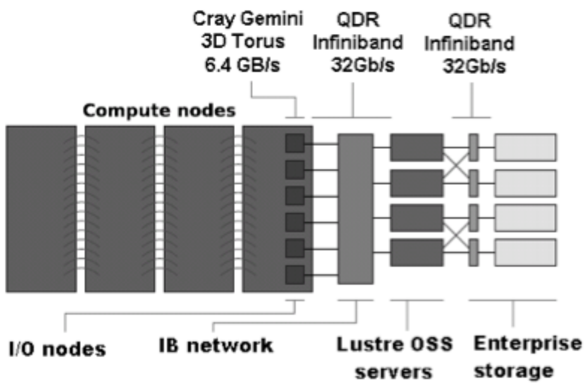


Figure 10: Cray Cielo XE6 I/O system

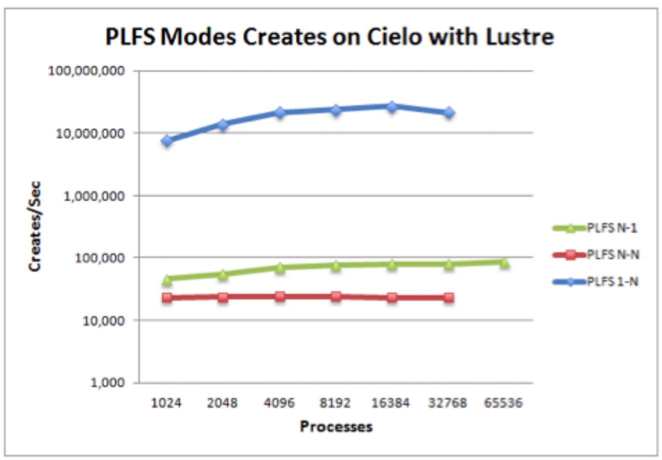


Figure 11: PLFS Modes Scalability with number of processes

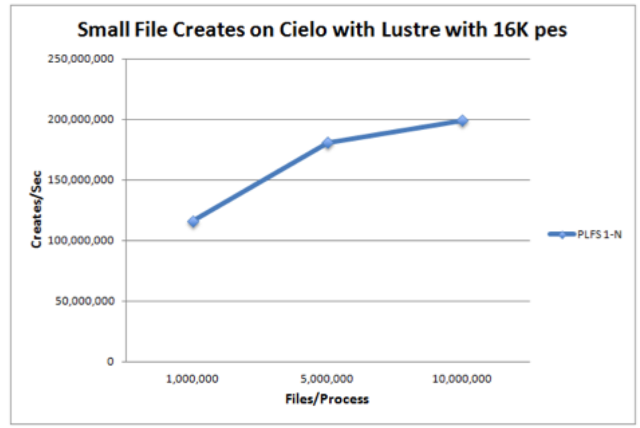


Figure 12: Small Files Scalability for files per process

Figures

Table I: PLFS modes versus HFS small files performance

Target	Files /proc	Files create	Time seconds	files/s	Percent
Mac HFS	3000	45000	17.1	2627	100
N-1	3000	45000	159.2	282	11
N-N	3000	45000	101.0	445	17
1-N	3000	45000	4.2	10704	407
Mac HFS	10000	150000	50.8	2948	112
1-N	10000	150000	10.3	14516	553
1-N	100000	1500000	78.1	19193	731

Table II: PLFS versus Linux FS small files

operation	ext2	ext3	ext4	reiserFS	JFS	PLFS
creates/s in 1 dir	204	141	324	401	108	282(N-1)
creates/s in 10 dir	990	961	1000	491	47	445(N-N)

Table III: PLFS small file versus Lustre on Linux cluster

Target	Files /proc	Files create	Time sec	files/s	Ratio
1-N Creates	10,000	10,240,000	13.5	7,573,964	1815
1-N Write 1b	10,000	10,240,000	36.6	2,797,814	670
Lustre Creates	200	204,800	49.1	4173	1
Lustre Write 1b	200	204,800	73.2	2798	0.68